



DSpace How-To Guide

Tips and tricks for managing common DSpace chores

(Now serving DSpace 1.4.2 and Manakin 1.1)

Authors / Contributors

(alphabetically by last name)

Tim Donohue

Scott Phillips

Dorothea Salo

© 2006-2007 by all Authors / Contributors (listed above)

This work is licensed under Creative Commons Attribution-Noncommercial-Share Alike 3.0 License.

To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/>

Introduction

This short booklet is intended to introduce the commonest non-obvious customization-related tasks for newcomers to DSpace administration. It has been written against the stable version 1.4.2 of DSpace and the Manakin user interface, version 1.1.

We have tried to include instructions for different operating systems as required; most customizations, however, work identically cross-platform.

Before you start

Different parts of DSpace live in different areas on the DSpace server. Because each DSpace administrator decides where some parts of DSpace live, and operating systems have different ideas about where other parts live, we have employed italics to mark missing pieces of file paths. Some directories whose precise location varies among systems require special attention:

- *[dspace-source]* – The directory into which the DSpace administrator unpacks the downloaded DSpace code.
- *[manakin-source]* – The directory into which the DSpace administrator unpacks the downloaded Manakin source code.
- *[dspace]* – The directory into which the Java `ant` command deposits compiled DSpace code. This location is set in DSpace's configuration file, `dspace.cfg`, as `dspace.dir`
- *[tomcat]* – The directory where Apache Tomcat is installed. If you are not using Apache Tomcat, you may wish to customize some of these “how-to” documents to better match your servlet engine of choice (e.g. Jetty, Resin, etc).

Other DSpace resources

Although this “DSpace How-To Guide” introduces many common customizations currently available within DSpace, it should not be considered a stand-alone document. You should be aware of many other valuable DSpace resources, including:

- DSpace Homepage – <http://www.dspace.org>
- DSpace Wiki – <http://wiki.dspace.org/>
- DSpace Technical FAQ – <http://wiki.dspace.org/TechnicalFAQ>
- DSpace Resources Listing (including System Documentation and Mailing Lists) – <http://wiki.dspace.org/DspaceResources>
- Manakin documentation – <http://di.tamu.edu/projects/xmlui/>

Other useful resources

This “DSpace How-To Guide” introduces many customizations which rely on some general knowledge of other technologies, including XHTML, CSS, XML and XSLT. If you need a refresher on any of these technologies, here are a few useful web resources (available as of June 2007):

- W3Schools Tutorials/Guides (<http://www.w3schools.com/>) – introductory tutorials, references, and examples for XHTML, CSS, XML, XSLT, SQL, among many others.
- HTML Dog (<http://www.htmldog.com/>) – HTML and CSS References and tutorials for all levels of knowledge
- Holy CSS Zeldman! (<http://www.dezwozhere.com/links.html>) – all things CSS, with intermixed links to good HTML, Javascript, and web design sites.
- MaxDesign (<http://css.maxdesign.com.au/>) - CSS tutorials for web developers. Covers customizing page layout using CSS-based floats, lists and selectors.

Other useful skills

Many tasks that are cumbersome to manage through the DSpace web interface can be easily managed in the database with a little SQL. For example, withdrawing all the items in a particular collection takes quite some time in DSpace proper, but can be done in just one quick SQL query. However, you should always be sure to backup your database before running any SQL queries which will modify or remove a large number of items! Time spent learning SQL basics will reap rich returns in time saved.

Table of Contents

Introduction.....	2	Change a form value.....	30
Table of Contents.....	4	Use Controlled Vocabularies.....	31
Quick Subject Index.....	5	Use Controlled Vocabularies.....	32
CUSTOMIZING DSPACE, VERSION 1.4.2.....	7	Change displayed item metadata.....	33
Rebuild DSpace.....	8	Use DSpace Statistics.....	34
Rebuild DSpace.....	9	Use DSpace Statistics.....	35
Change page text.....	10	Use Dspace Statistics.....	36
Add new text to a JSP.....	11	Use DSpace Statistics.....	37
Fix ???some.key.name???.....	12	Use the OAI-PMH interface.....	38
Change overall layout.....	13	Configure RSS feeds.....	39
Change single page layout.....	14	Configure RSS feeds.....	40
Add a custom metadata schema.....	15	Configure checksum checker.....	41
Add a new metadata field.....	16	Configure checksum checker.....	42
Move a metadata field.....	17	Configure checksum checker.....	43
Modify search fields.....	18	Troubleshoot an error.....	44
Modify search fields.....	19	CUSTOMIZING MANAKIN, VERSION 1.1.....	45
Modify default search options.....	20	Rebuild Manakin.....	46
Re-index DSpace.....	21	Rebuild Manakin.....	47
Configure media filters.....	22	Install a theme.....	48
Configure media filters.....	23	Create a new theme.....	49
Configure full text indexing.....	24	Customize header or footer.....	50
Modify browse fields.....	25	Modify item metadata display.....	51
Change browse layout.....	26	Add logos to an item's display.....	52
Alter submission input forms.....	27	Add text to a page (i18n).....	53
Alter submission input forms.....	28	Install an Aspect.....	54
Alter submission input forms.....	29	Troubleshoot an error.....	55

Quick Subject Index

General Tips

- Rebuild DSpace [8]
- Rebuild Manakin [46]
- Troubleshoot an error (DSpace) [44]
- Troubleshoot an error (Manakin) [55]

User Interface Tips

- *DSpace JSP User Interface*
 - Change overall layout [13]
 - Change single page layout [14]
 - Change page text [10]
 - Add new text to a JSP [11]
 - Fix ???some.key.name??? [12]
 - (See specific DSpace functionality for additional tips)
- *Manakin XML User Interface*
 - Install a theme [48]
 - Create a new theme [49]
 - Customize header or footer [50]
 - Modify item metadata display [51]
 - Add logos to an item's display [52]
 - Add text to a page (i18n) [53]
 - Install an Aspect [54]

Metadata Tips

- Add a custom metadata schema [15]
- Add a new metadata field [16]
- Move a metadata field [17]
- Change displayed item metadata [33]
- Use Controlled Vocabularies [31]

Submission Tips

- Alter submission input forms [27]
- Change a form value [30]
- Use Controlled Vocabularies [31]

Search Tips

- Modify default search options [20]
- Modify search fields [18]
- Configure full text indexing [24]
- Configure media filters [22]
- Re-index DSpace [21]

Browse Tips

- Change browse layout [26]
- Modify browse fields [25]

Management Tips

- Configure checksum checker [41]
- Use DSpace Statistics [34]

Dissemination Tips

- Configure RSS feeds [39]
- Use the OAI-PMH interface [38]

[This page intentionally left blank.]

Customizing DSpace, Version 1.4.2

Rebuild DSpace

Directories:

- `[dspace-source]`
- `[dspace-source]/build/`
- `[Tomcat]/webapps/` (Mac OSX Server: `/library/jboss/3.2/deploy`)
 - Note: JBOSS comes pre-installed with Mac OS X server. However, for both sever and desktop editions Tomcat may be used as with other platforms.

Quick Restart *(Just restarts the web server after configuration changes*)*

(*Exception – Changes to `Messages.properties` always requires a rebuild!)

1. Stop Tomcat (WARNING: this will bring down the website)
 - (Linux / OS X / Solaris) `[Tomcat]/bin/shutdown.sh`
 - (Mac OS X Server) Use Server Admin to stop Tomcat (“Application Server”)
 - (Windows) Use Tomcat Service Monitor (in Notification Area) to stop Tomcat
2. Start Tomcat
 - (Linux / OS X / Solaris) `[Tomcat]/bin/startup.sh`
 - (Mac OS X Server) Use Server Admin to start Tomcat (“Application Server”)
 - (Windows) Use Tomcat Service Monitor (in Notification Area) to start Tomcat

Quick Build: *(Quick build after smaller, usually JSP-based, changes)*

1. Logon to the server DSpace is running on (e.g. ssh). Make sure to login as the user who initially installed DSpace!
2. Open a command prompt (if you don’t have one already), and `cd [dspace-source]`
3. `ant update` (recompiles all DSpace code and reinstalls third-party JAR files)
4. Alternatively, if you do not need to reinstall JAR files, you could instead run `ant build_wars` (which just recompiles DSpace code)
5. `cp build/*.war [Tomcat]/webapps/`
 - (Mac OS X Server) `cp build/*.war /library/jboss/3.2/deploy`
6. Test changes in DSpace

Rebuild DSpace

(continued)

Full Refresh/Rebuild: *(Completely refresh all of DSpace)*

1. Logon to the server DSpace is running on (e.g. ssh). Make sure to login as the user who initially installed DSpace!
2. Open a command prompt (if you don't have one already), and `cd [dspace-source]`
3. `ant clean` (removes old compiled code)
4. `ant update` (recompiles all DSpace code and reinstalls third-party JAR files)
5. Stop Tomcat (*WARNING: this will bring down the website*)
 - (Linux / OS X / Solaris) `[Tomcat]/bin/shutdown.sh`
 - (Mac OS X Server) Use Server Admin to stop Tomcat ("Application Server")
 - (Windows) Use Tomcat Service Monitor (in Notification Area) to stop Tomcat
6. `cp build/*.war [Tomcat]/webapps/`
 - (Mac OS X Server) `cp build/*.war /library/jboss/3.2/deploy`
7. (Optional, but recommended) Completely remove any `dspace` or `dspace-oai` directories created in `[Tomcat]/webapps` (`/library/jboss/3.2/deploy` for Mac) - BE CAREFUL...you don't want to remove the wrong directory!
 - To force Tomcat to recompile everything, you may also wish to remove any `dspace` or `dspace-oai` directories created in `[Tomcat]/work/Catalina/localhost`
8. Start Tomcat
 - (Linux / OS X / Solaris) `[Tomcat]/bin/startup.sh`
 - (Mac OS X Server) Use Server Admin to start Tomcat ("Application Server")
 - (Windows) Use Tomcat Service Monitor (in Notification Area) to start Tomcat
9. Test your changes in DSpace

Change page text

Files:

- `[dspace-source]/jsp/` (JSP containing the text you want to change)
- `[dspace-source]/config/language-packs/Messages.properties`

Instructions:

1. Open `Messages.properties` and search for the text you wish to change.
 - Note: `Messages.properties` contains pairs of “keys” and “values”. For example:
`|jsp.home.search1 = Search`
 Generally speaking, the “key” usually refers to the location of the JSP on which this text resides (e.g. `jsp.home.search1` is “search-related” text displayed in `[dspace-source]/jsp/home.jsp`)
2. If `Messages.properties` contains that text in more than one place, open the relevant JSP and find the `key` attribute of the appropriate `<fmt:message>` element. For example:
`|<fmt:message key="jsp.home.search1" />`
3. Change the text (that corresponds to the key) in `Messages.properties`.
4. Perform the steps in **Rebuild DSpace**.

Notes:

1. When adding or modifying text in `Messages.properties`, be very careful that you have automatic *word-wrap* turned off in your text editor! The “key” and its corresponding “value” must always be on the same line within `Messages.properties`
 - a. (e.g.) This is **NOT** a valid entry in `Messages.properties`:

```
|jsp.community-home.heading1 = This is a really long heading
|which actually gets wrapped automatically by my text editor
|so that it ends up on three separate lines.
```

Add new text to a JSP

Files:

- `[dspace-source]/jsp/` (JSP containing the text you want to change)
- `[dspace-source]/config/language-packs/Messages.properties`

Instructions:

1. Open the relevant JSP and add a new `<fmt:message>` element where you want the new text to appear, making sure that the value of the `key` attribute is unique. For example:

```
| <fmt:message key="jsp.home.mynewkey" />
```

2. Open Messages.properties and (for neatness) find the list of key values for the JSP you just changed. Add a new line corresponding to your new key.

```
| jsp.home.mynewkey = Text of my new key
```

3. Note: Messages.properties contains pairs of “keys” and “values”. For example:

```
| jsp.home.search1 = Search
```

Generally speaking, the “key” usually refers to the location of the JSP on which this text resides (e.g. `jsp.home.search1` is “search-related” text displayed in `[dspace-source]/jsp/home.jsp`)

4. Perform the steps in **Rebuild DSpace**.

Notes:

- You can, of course, simply add the text directly to the JSP, but you will find it easier to maintain text in your DSpace installation if it is all kept in Messages.properties.
- Remember, when adding or modifying text in Messages.properties, be very careful that you have *word-wrap turned off* in your text editor!

Fix ???some.key.name???

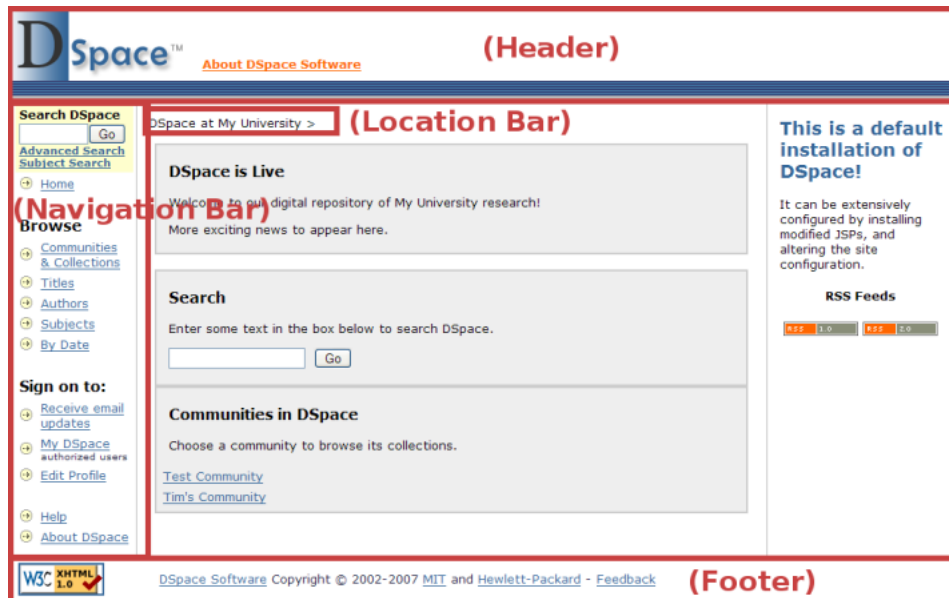
File:

- *[dspace-source]/config/language-packs/Messages.properties*
- *[dspace-source]/jsp/local/ (JSP producing ???some.key.name??? text)*

Instructions:

1. Search for the string inside the question marks in the Messages.properties file.
2. Search for the same string inside the JSP; it should be the value of a **key** attribute to a `<fmt:message>` element. If both the JSP and Messages.properties appear to contain the string, be sure the strings match exactly.
3. If Messages.properties does not contain the string, add it:
| `some.key.name=Text that should appear`
4. Perform the steps in **Rebuild DSpace**.

Change overall layout



Files:

- `[dspace-source]/jsp/local/layout/*.jsp`
- `[dspace-source]/jsp/local/styles.css.jsp`

Instructions:

1. Change the HTML in header-default.jsp (Default Header) , footer-default.jsp (Default Footer), location-bar.jsp (Location Bar), navbar.jsp (Default Navigation Bar), and navbar-admin.jsp (Admin Navigation Bar).
2. Change the CSS in styles.css.jsp.
3. Perform the steps in **Rebuild DSpace**.

Notes:

- Be careful of moving the search form (in the navigation bar) earlier in the page (e.g. to page-top). This can break the e-person selector in the Administration user interface. Check the DSpace Technical FAQ (<http://wiki.dspace.org/TechnicalFAQ>) for possible fixes.

Change single page layout

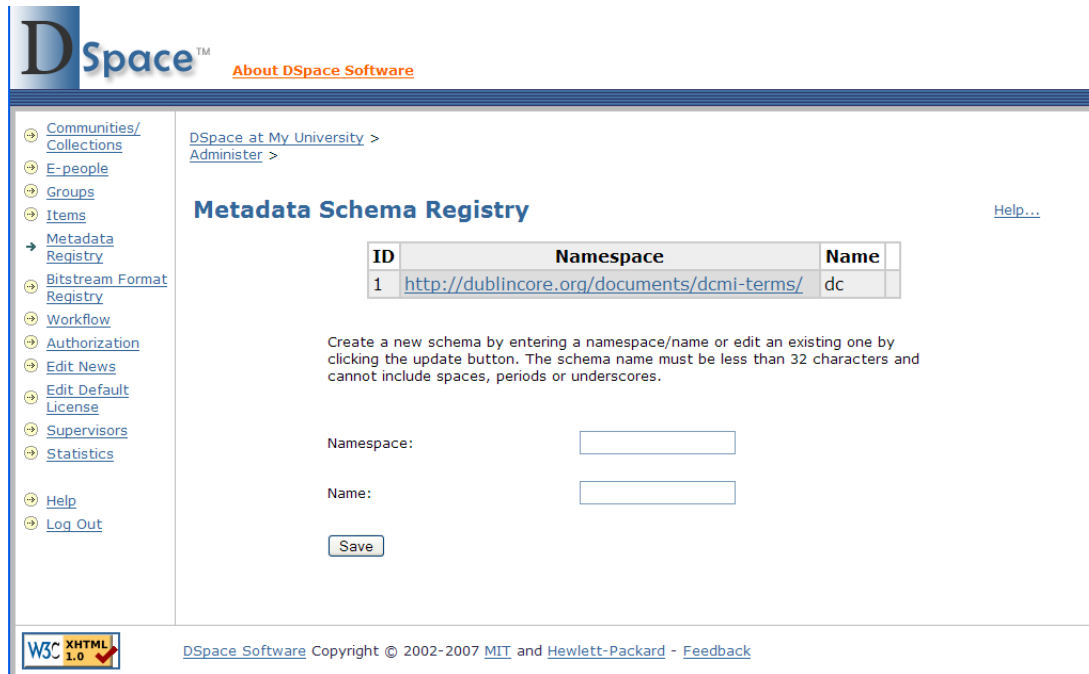
Files:

- Any JSP in *[dspace-source]*/jsp/local/

Instructions:

1. Find the following JSP Tag near the top of the JSP:
`<dspace:layout ... >`
2. Pass this tag any of the following “attributes”, to customize the layout of this JSP:
 - **style** – specifies the “style” of header/footer to display for this JSP
 - (e.g.) **style="myStyle"** means the header-*myStyle*.jsp and footer-*myStyle*.jsp will be used for this JSP.
 - If **style** is unspecified, header-default.jsp and footer-default.jsp are used.
 - **navbar** – specifies the navigation bar to use for this JSP
 - (e.g.) **navbar="myNavigation"** means the navbar-*myNavigation*.jsp will be used for this JSP
 - **navbar="off"** turns off the navigation bar on a page.
 - If **navbar** is unspecified, navbar-default.jsp is used.
 - **locbar** – specifies type of location bar to use. There are only a few values of real importance:
 - **locbar="off"** – turns off the location bar on this JSP.
 - **locbar="noLink"** – do not provide links in location bar.
 - **locbar="commLink"** – attempt to provide all parent communities within the location bar.
 - If **locbar** is unspecified, all parent communities/collections are displayed as links in the location bar.
3. Perform the steps in **Rebuild DSpace**.

Add a custom metadata schema



Files:

- <http://web-address-to-my-dspace/dspace-admin> (Requires Administrator Login)

Instructions:

1. Login as a DSpace Administrator and visit the DSpace Administration user interface (<http://web-address-to-my-dspace/dspace-admin>)
2. Click on the “Metadata Registry” in order to see all current metadata schemas within DSpace. By default, you should only see the Dublin Core (dc) schema
3. At the bottom of the page, enter in a new metadata schema “namespace” and give it a “name”. (There is no requirement that the namespace be resolvable. Also, you may wish the “name” to be only a few characters, since it serves dual purpose as a prefix for this schema). Click “Save”
4. The new metadata schema is now added to the underlying database. You will then want to add new metadata fields by following the instructions in [Add a new metadata field](#). It is also possible to move metadata fields from one schema to another, by following the instructions in [Move a metadata field](#).

Notes:

- DSpace currently only supports “flat” (non-hierarchical) metadata schemas.

Add a new metadata field

Add Metadata Field

To create a new field you must provide a unique element and qualifier pair. The qualifier may be left blank if desired and the element and qualifier cannot contain spaces, underscores or periods.

Element:

Qualifier:

Scope Note:

Files:

- <http://web-address-to-my-dspace/dspace-admin> (Requires Administrator Login)

Instructions:

1. Login as a DSpace Administrator and visit the DSpace Administration user interface (<http://web-address-to-my-dspace/dspace-admin>)
2. Click on the “Metadata Registry” in order to see all current metadata schemas within DSpace. Then, click on the namespace of the schema to add a field to that particular schema.
3. At the bottom of the page, you will find the *Add Metadata Field* form, where you can specify an “element”, “qualifier”, and “scope note” for the new metadata field. Only the “element” is required. Click “Add New” to add the new field.
4. The new metadata field is now added to the underlying database as a member of the current metadata schema. If you wish, you can now make this field searchable (see [Modify search fields](#)), add this field to the submission forms(see [Alter submission input forms](#)), and/or display this field in the item display (see [Change displayed item metadata](#)).

Move a metadata field

Move Field

To move one or more field, select the desired fields (use control-key to select multiple) and the target schema. If the target schema already has fields with identical names the fields will not be moved.

Element:

Schema:

Files:

- <http://web-address-to-my-dspace/dspace-admin> (Requires Administrator Login)

Instructions:

1. Login as a DSpace Administrator and visit the DSpace Administration user interface (<http://web-address-to-my-dspace/dspace-admin>)
2. Click on the “Metadata Registry” in order to see all current metadata schemas within DSpace. Then, click on the namespace of the schema you wish to move a metadata field *from*.
3. At the bottom of the page, you will find the *Move Field* form. Using this form you can select one or more fields (in the current schema), and select the schema you wish to move those fields to. Once you have selected the field(s) and the schema to move them to, click “Move”.
4. The selected metadata fields are now moved to the selected schema. Since you moved these fields, you may need to update how they are being used in DSpace. For example, you can now make these fields searchable (see [Modify search fields](#)), add these fields to the submission forms (see [Alter submission input forms](#)), and/or display these fields in the item display (see [Change displayed item metadata](#)).

Notes:

- Moving a metadata field will also *remove* it from its existing metadata schema. However, any items which previously used this metadata field will be updated automatically (i.e. no existing metadata should be lost).

Modify search fields

Files:

- `[dspace]/config/dspace.cfg`
- `[dspace-source]/jsp/local/search/advanced.jsp`
- `[dspace-source]/config/language-packs/Messages.properties`

Instructions:

1. Look for this line in `dspace.cfg`:

```
##### Fields to Index for Search #####
```

2. Beneath it you will see several lines like this:

```
search.index.1 = author:dc.contributor.*
search.index.2 = author:dc.creator.*
search.index.3 = title:dc.title.*
search.index.4 = keyword:dc.subject.*
search.index.5 = abstract:dc.description.abstract
```

3. Add another `search.index.#` line to the bottom. If you just want to add a different Dublin Core (`dc`) field to one of the existing “named indices”, use the models above as a guide. If you wish to specify a different metadata schema, replace `dc` with the other metadata schema's name.

- The “name” to the left of the colon (e.g. `author`, `title`, `keyword`, etc) is *important*. In the above example, a search on “author” is specified to search all Dublin Core (`dc`) `contributor` and `creator` fields. Whereas, a search on “abstract” only searches the `dc.description.abstract` field.

4. If you want to add an entirely new search field, you will also have to modify `Messages.properties` (see [Change page text](#)) to add a user-friendly label for it, and the advanced-search JSP (`advanced.jsp`) to add an appropriate `<option>` element, as below:

```
<option value="author" <%= field1.equals("author") ?
"selected=\"selected\" " : "" %>><fmt:message
key="jsp.search.advanced.type.author"/></option>
```

- The “value” attribute of your `<option>` element should correspond to the name of one of your search indices (e.g. `author`, `title`, `keyword`, etc.)
5. Perform the steps in [Re-index DSpace](#).
 6. Perform the steps in [Rebuild DSpace](#).

Modify search fields

(continued)

Notes:

- In DSpace, the most confusing concept regarding search fields is the keyword search.
 - In the basic search boxes (as seen below), any terms entered are searched for *anywhere* within any of the search indices (i.e. any of the `search.index.#` fields in `dspace.cfg`), or the full text of the document (if it is full-text indexable). These search boxes perform what most refer to as a *keyword* or *keyterm* search.



- However, to make things a little confusing, you'll notice a keyword search index listed in `dspace.cfg`:

| `search.index.4 = keyword:dc.subject.*`

This (rather inappropriately named) index is actually used during *subject* specific searches (hence the `subject.*`). It does **not** have any control over a normal *keyword* search that is run from the basic search box in DSpace.

Modify default search options

Files:

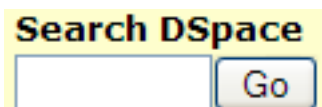
- `[dspace]/config/dspace.cfg`

Instructions:

1. You may wish to customize which boolean search operator is used by default. The DSpace search engine defaults to using boolean **OR** (requires at least one search term to be present) when multiple terms are entered. However, if desired, you may specify to use boolean **AND** (requires all terms to be present) by modifying the following line in `dspace.cfg`:

| `search.operator = OR`

The specified search operator will be the default operator used whenever someone enters multiple terms into the main search box



2. If you want to modify which metadata fields are searchable, see [Modify search fields](#)

Re-index DSpace

Files:

- `[dspace]/bin/index-all`

Instructions:

1. Log on to the machine running DSpace.
2. Change directory to `[dspace]/bin`.
3. Run the re-indexing script
 - (Linux or Mac OSX) `./index-all`
 - (Windows) *run from command prompt*

```
./dsrun org.dspace.browse.InitializeBrowse  
./dsrun org.dspace.search.DSIndexer
```
4. Stop and restart Tomcat (See *Quick Restart* in **Rebuild DSpace**).

Notes:

- This process re-creates DSpace's search and browse indices. Run it after anything you do to the install that could change the content of these indices (e.g. manually changing metadata, withdrawing items). In addition, run it if you decide to change your searchable metadata fields (see **Modify search fields**).
- You may wish to have a scheduled process (e.g. `cron`) to re-index DSpace daily. Lots of little changes that add up over time without a re-indexing can cause DSpace's search function to become erratic.

Configure media filters

Files:

- `[dspace]/bin/filter-media`
- `[dspace]/config/dspace.cfg`
- A scheduling program (e.g. Unix `cron` or Windows Scheduled Tasks)

Instructions:

1. In DSpace, “media filters” are what control both *full-text indexing* and *automated creation of thumbnail images*. Both can be scheduled by calling the `filter-media` script (which in turn calls the `org.dspace.app.mediafilter.MediaFilterManager` class).
2. The list of all currently enabled “media filters” is available in your `dspace.cfg` configuration file under the section labeled:

```
##### Media Filter plugins (through PluginManager) #####
```

3. If you wish to disable or enable a specific “media filter”, you can remove or add them from the sequence list in that section:

```
plugin.sequence.org.dspace.app.mediafilter.MediaFilter = \
    org.dspace.app.mediafilter.PDFFilter, \
    org.dspace.app.mediafilter.HTMLFilter, \
    org.dspace.app.mediafilter.WordFilter, \
    org.dspace.app.mediafilter.JPEGFilter
```

(Make sure that each separate line ends with a backslash `\` character! A list of what each “media filter” does is available in the *Notes* section on the next page)

4. For Linux or Mac OSX, you can schedule the `filter-media` shell script to run by adding a `cron` entry similar to the following to the crontab for the user who installed DSpace:

```
| 0 2 * * * [dspace]/bin/filter-media
```

(The above entry would schedule filter-media to run nightly at 2am. You would need to change `[dspace]` to the full path of your DSpace installation directory.)

5. For Windows, you will be unable to use the `filter-media` shell script. Instead, you should use Windows Scheduled Tasks to schedule the following command to run at the appropriate time of day:

```
| [dspace]/bin/dsrun.bat
| org.dspace.app.mediafilter.MediaFilterManager
```

(The above command should appear on a single line.)

Configure media filters

(continued)

Notes:

- Below is a listing of all currently available Media Filters, and what they actually do:
 - **HTMLFilter** - extracts the full text of HTML documents for full text indexing.
 - **JPEGFilter** - creates thumbnail images of GIF, JPEG and PNG files
 - **BrandedPreviewJPEGFilter** - creates a branded preview image for GIF, JPEG and PNG files (*disabled by default*)
 - **PDFFilter** - extracts the full text of Adobe PDF documents (only if text-based or OCR'd) for full text indexing
 - **WordFilter** - extracts the full text of Microsoft Word or Plain Text documents for full text indexing
- Please note that the **filter-media** or **MediaFilterManager** will automatically update the DSpace search index by default (see **Re-index DSpace**). This is the recommended way to run these scripts. But, should you wish to disable it, you can pass the **-n** flag to either script to do so.
- The following additional options are also available for either the **filter-media** or **MediaFilterManager** scripts:
 - **-v** = verbose mode (prints out all extracted text and additional messages)
 - **-f** = forces reprocessing of all bitstreams (by default only unprocessed bitstreams are processed by this script)
 - **-n** = do not update the search index after completion
 - **-i <handle>** = only process bitstreams within a particular community/collection/item represented by the **<handle>**
 - **-m <#>** = only process a maximum number of bitstreams (specified by **<#>**) . This is useful if you want to process bitstreams little-by-little in order to avoid taxing your server too much.

Configure full text indexing

Files:

- `[dspace]/bin/filter-media`
- `[dspace]/config/dspace.cfg`

Instructions:

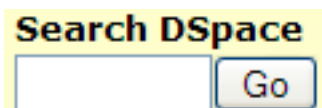
1. First, follow the instructions in Configure media filters, since full text indexing is performed by those media filters (specifically the `HTMLFilter`, `PDFFilter`, and `WordFilter`)
2. You may wish to modify the `search.maxfieldlength` field in your `dspace.cfg` configuration file. This field specifies the maximum number words to index for each document, and by default is set to the first 10,000 words. (Set it to the value `-1`, if you want to index an unlimited number of words)

| `search.maxfieldlength = 10000`

If you choose to modify the `search.maxfieldlength` field you must **Re-index DSpace** before the change will take affect.

Notes:

- Full text indexing is only available for the following formats at this time:
 - Adobe PDF (only if text-based or OCR'd)
 - Microsoft Word
 - Plain Text
 - HTML
- Full text searching in DSpace occurs when a user searches via the default search box (see below), or when a user selects the “Keyword” option from the Advanced Search screen.



Modify browse fields

Files:

- `[dspace]/config/dspace.cfg`

Instructions:

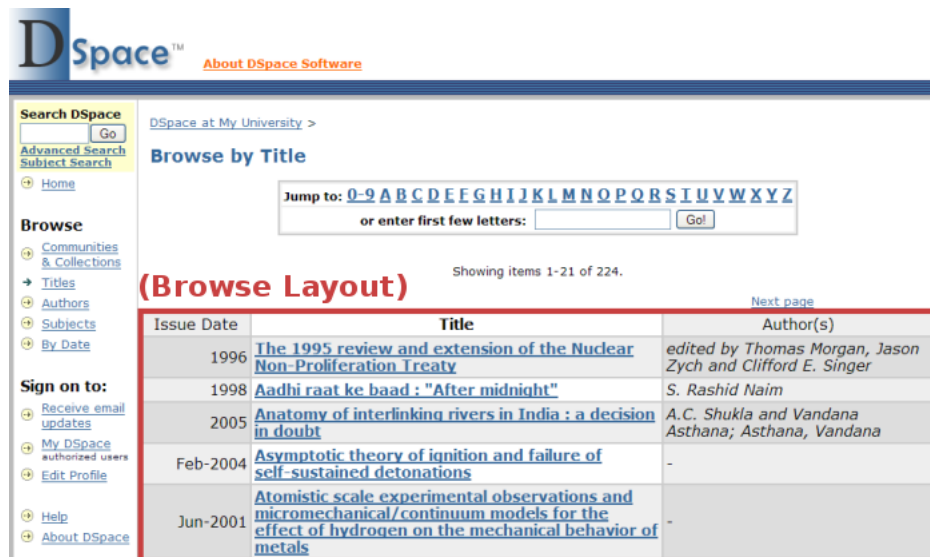
1. In `dspace.cfg`, you may uncomment the following lines to specify which metadata fields you would like to be browsing in Dspace's "Browse by Date", "Browse Authors", "Browse Titles" and "Browse Subjects" interfaces, respectively:

```
# webui.browse.index.date = dc.date.issued
# webui.browse.index.author = dc.contributor.*
# webui.browse.index.title = dc.title
# webui.browse.index.subject = dc.subject.*
```
2. The above commented out values of these fields show the default fields browsed within DSpace. Currently, you are only able to specify a single metadata element (e.g. `dc.date`) for each index, but may add one or more qualifiers (e.g. the `issued` qualifier on `dc.date`). Note the following about qualifiers:
 - Adding an asterisk (*) in place of a qualifier allows you to specify all possible qualifiers. So, specifying `dc.subject.*` means the browse index would include all of the following fields: `dc.subject`, `dc.subject.lcsh`, `dc.subject.custom`, etc.
 - You can also specify no qualifier (e.g. `dc.title`), which means the browse index does *not* include fields with qualifiers (so specifying `dc.title` will *not* include `dc.title.alternative`)
3. If you modify any of the browse indices in `dspace.cfg`, you must Re-index DSpace for the changes to take affect.

Notes:

- You can also customize which metadata fields are displayed whenever you browse a list of items (through any of the DSpace browse interfaces). To do so, see the section entitled **Change browse layout**.

Change browse layout



Files:

- `[dspace]/config/dspace.cfg`

Instructions:

1. In `dspace.cfg`, you can change the metadata fields which are displayed for each item in the browse item listing. To do so, look for the following single line in your `dspace.cfg` and uncomment it:


```
#webui.itemlist.columns = dc.date.issued(date),
dc.title, dc.contributor.*
```
2. By default, whenever you see a list of items, you'll see the *Issue Date*, *Title*, and *Author(s)* columns in that order. However, you can use the above configuration option to rearrange these columns or change which columns you want displayed. Although not required, you may wish to match these columns with your specified browse indices (see [Modify browse fields](#)).
3. As normal, specifying a single qualifier (e.g. `issued` on `dc.date`) or no qualifier (e.g. `dc.title`) will only display values from those specific metadata fields. However, specifying an asterisk (*) as the qualifier will display all fields for that element (whether it has a qualifier or not). So, specifying `dc.contributor.*` will cause any/all of the following fields to be displayed in that column: `dc.contributor`, `dc.contributor.author`, `dc.contributor.editor`, etc.
4. Stop and restart Tomcat (See *Quick Restart* in [Rebuild DSpace](#)), for the changes to take affect.

Alter submission input forms

Submit: Describe this Item

Please fill in the requested information about this submission below. In most browsers, you can use the tab key to move the cursor to the next input box or button, to save you having to use the mouse each time. ([More Help...](#)) **(Submission Form)**

Enter the names of the authors of this item below.

Last name	First name(s) + "Jr"
e.g. <i>Smith</i>	e.g. <i>Donald Jr</i>

Authors

Enter the main title of the item.

Title

Enter the series and number assigned to this item by your community.

Series Name	Report or Paper No.

Series/Report No.

Files:

- `[dspace]/config/input-forms.xml`

Instructions:

1. This XML file contains form definitions, each contained within its own `<form>` element. Each `<form>` element contains `<page>` elements that represent single pages of input. Each `<page>` element contains `<field>` elements with instructions for creating the fields on that page. This structure is mocked up below, for a single form with a single page.

```
<form-definitions>
  <form name="traditional">
    <page number="1">
      <field>
        ...
      </field>
    </page>
  </form>
</form-definitions>
```

Alter submission input forms

(continued)

2. The default form has a `name` attribute whose value is “`traditional`”. To change all submission forms throughout DSpace, edit the XML inside this form.
3. To change a submission form *for a specific collection*, add a new `<form>` element with a new and unique collection-related value for the `name` attribute. (You can copy the XML from the default and edit as desired, as long as you change the `name` attribute value.)
 - Once you’ve created your new `<form>` element, you will have to map it to your specific collection using the `<form-map>` section at the top of the XML:

```
<form-map>
  <name-map collection-handle="default" form-
name="traditional" />
</form-map>
```

- You’ll need to create a new `<name-map>` element to “map” your collection’s handle to the new `<form>` you just created.
4. Create all appropriate `<page>` elements, numbering them sequentially starting from 1 in the `number` attribute. Within them, add individual `<field>` elements according to this template:

```
<field>
  <dc-schema>dc</dc-schema>
  <dc-element>identifier</dc-element>
  <dc-qualifier>citation</dc-qualifier>
  <repeatable>>false</repeatable>
  <label>Citation</label>
  <input-type>onebox</input-type>
  <hint>Enter the standard citation for the previously
issued instance of this item.</hint>
  <required></required>
  <vocabulary></vocabulary>
</field>
```

5. Stop and restart Tomcat (See *Quick Restart* in **Rebuild DSpace**).

Alter submission input forms

(continued)

Notes:

- The `<required>` element contains the textual hint displayed to the submitter about why the field is required. Leave it empty for optional fields.
- The `<vocabulary>` element is *optional*. It allows you to specify the controlled vocabulary (see [Use Controlled Vocabularies](#) for more information) that this field should select its values from. This field also has an optional `closed` attribute. If `closed` is set to `true`, a user can only select values from the controlled vocabulary. By default, `closed` is set to `false`, which allows a user to also enter in free text if he/she chooses. For example:

```
| <vocabulary closed="true">srsc</vocabulary>
```

The name of the controlled vocabulary must correspond to the name of the XML file (without “.xml”) which contains the vocabulary. So, in the above example, `srsc` references the vocabulary specified in the file located at `[dspace]/config/controlled-vocabularies/srsc.xml`

- Valid input types (for `<input-type>`) are:
 - “date”
 - “name” (two text boxes, labeled last and first name)
 - “onebox” (a one-line textbox)
 - “twobox” (two textboxes on a single line)
 - “dropdown” (for which you must specify a `value-pairs-name` attribute referring to the `<value-pairs>` list of allowed values, see [Change a form value](#))
 - “qualdrop_value” (a textbox, which is preceded by a “qualifying” dropdown of values. Requires a `value-pairs-name` attribute, similar to “dropdown”. Also requires `<repeatable>` is set to “true”)
 - “textarea”
 - Setting the `<repeatable>` element to “true” creates an “Add more” button, which allows you to add multiple values into that particular field. Examples of this include the authors and keywords fields in the standard DSpace submission process.

Change a form value

Submit: Describe this Item

Please fill in the requested information about this submission below. In most browsers, you can use the tab key to move the cursor to the next input box or button, to save you having to use the mouse each time. ([More Help...](#))

Enter the names of the authors of this item below.
Last name e.g. *Smith* First name(s) + "Jr" e.g. *Donald Jr*

Authors

Enter the main title of the item.

Title

Enter the series and number assigned to this item by your community.

Series Name Report or Paper No.

Series/Report No.

If the item has any identification numbers or codes associated with it, please enter the types and the actual numbers or codes below.

Identifiers

Select the type(s) of content of the item. To select more than one value in the list, you may have to hold down the "CTRL" or "Shift" key.

Type

Select the language of the main content of the item. If the language does not appear in the list below, please select "Other". If the content does not really have a language (for example, if it is a dataset or an image) please select "N/A".

Language

(Fields with Form Values)

Files:

- `[dspace]/config/input-forms.xml`

Instructions:

1. Look for the `<form-value-pairs>` element in `input-forms.xml`.
2. Look for the `<value-pairs>` element for the Dublin Core element whose values you wish to change.
3. Add or change a `<pair>` element, where the `<displayed-value>` element contains what the submitter sees in the dropdown box, and the `<stored-value>` element contains what DSpace should archive as the information's label:

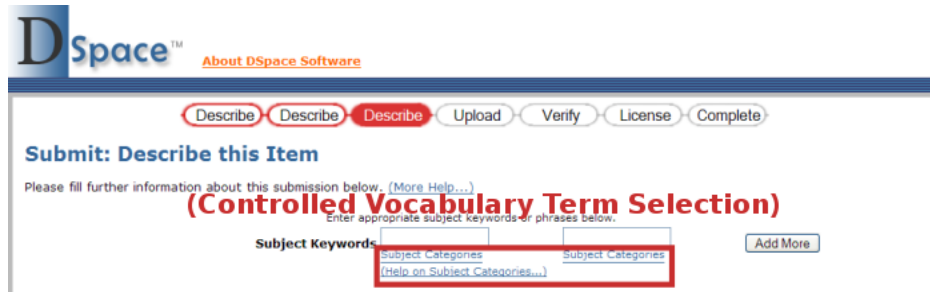
```
<pair>
  <displayed-value>Gov't Doc #</displayed-value>
  <stored-value>govdoc</stored-value>
</pair>
```

4. Stop and restart Tomcat (See *Quick Restart* in **Rebuild DSpace**).

Notes:

- This set of instructions changes the existing choices when DSpace displays an option-box or dropdown menu, as for the Dublin Core "type" and "language" elements. It does not add a new submission widget/field to the user interface; to do this, see **Alter submission input forms**.

Use Controlled Vocabularies



Files:

- `[dspace]/config/controlled-vocabularies/*`
- `[dspace]/config/dspace.cfg`

Instructions:

1. By default, the controlled vocabulary functionality is disabled since it is not compatible with [WAI Guidelines](#). To enable it, uncomment the following line in your `dspace.cfg`:

```
| #webui.controlledvocabulary.enable = true
```

2. DSpace only has the license to redistribute the following controlled vocabularies:

- Norwegian Science Index (`nsi.xml`) – By default, this vocab is not used
- Swedish Research Subject Categories (`srrsc.xml`) – When enabled, this vocab is used with the Subject (`dc.subject`) metadata field.

3. Each of the above controlled vocabularies is defined (in `[dspace]/config/controlled-vocabularies/`) via an XML with structure similar to:

```
<node id="acmccs98" label="ACMCCS98">
  <isComposedBy>
    <node id="A." label="General Literature">
      <isComposedBy>
        <node id="A.0" label="GENERAL"/>
        <node id="A.1" label="INTRODUCTORY AND SURVEY"/>
        ...
      </isComposedBy>
    </node>
    ...
  </isComposedBy>
</node>
```

4. The above XML structure would generate a hierarchical controlled vocabulary with the following structure:

Use Controlled Vocabularies

(continued)

ACMCCS98

General Literature

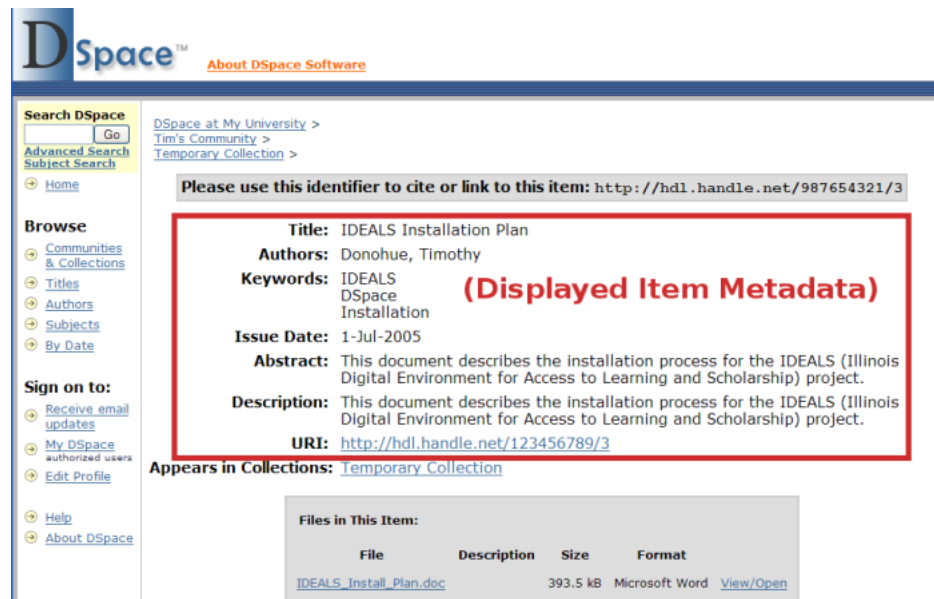
GENERAL

INTRODUCTORY AND SURVEY

Using this structure, if a submitter selected **GENERAL** in the above hierarchy, the item would be assigned a subject which is an aggregation of all terms above the selected term in the hierarchy, e.g. **ACMCCS98::General Literature::GENERAL**

5. You may create your own custom controlled vocabulary by generating an XML structure for it similar to above. Once created, place your custom controlled vocabulary alongside the preexisting controlled vocabularies in *[dspace]/config/controlled-vocabularies/*
 - There is a schema provided at *[dspace-source]/docs/controlledvocabulary.xsd*. The following fields are currently available in this XML Structure (*optional* attributes/fields are marked)
 - **<node>** - represents a single term in the hierarchy
 - **id** (*optional*) - a unique identifier for this term
 - **label** - a label for this term. This is the actual controlled text displayed in DSpace and stored in the metadata for an item.
 - **<isComposedBy>** (*optional*)- represents the “composed by” relationship between terms. So, in the example on the previous page, **General Literature** is specified as being composed by **GENERAL** and **INTRODUCTORY AND SURVEY**
 - **<isRelatedTo>** (*optional*) - similar to **<isComposedBy>**, but represents the “is related to” relationship between terms.
 - **<hasNote>** (*optional*) - specifies a note related to a term. This is never displayed but is useful for your own documentation/notes.
6. To actually use a controlled vocabulary with the submission input forms (see image on previous page), you must specify the controlled vocabulary to use for a specific metadata field. This can be achieved using the **<vocabulary>** tag within *input-forms.xml*. For more information, see the notes on the **<vocabulary>** tag in the section **Alter submission input forms**
7. Stop and restart Tomcat (See *Quick Restart* in **Rebuild DSpace**).

Change displayed item metadata



Files:

- `[dspace]/config/dspace.cfg`
- `[dspace-source]/config/language-packs/Messages.properties`

Instructions:

1. To change the metadata display, change the ordering of Dublin Core elements after `webui.itemdisplay.default` (in `dspace.cfg`), or add new elements:


```
webui.itemdisplay.default = dc.title, dc.title.alternative,
dc.contributor.*, dc.subject, dc.date.issued(date),
dc.publisher
```
2. To change the display name of a metadata field, search for its Dublin Core key name (which will begin with `metadata.`) in `Messages.properties`:


```
metadata.dc.title.alternative = Title (French)
```
3. Perform the steps in **Rebuild DSpace**.

Use DSpace Statistics



Files:

- `[dspace]/bin/stat-*` (i.e. all files beginning with “stat-”)
- `[dspace]/config/dspace.cfg`
- `[dspace]/config/dstat.cfg` and `dstat.map`
- <http://web-address-to-my-dspace/statistics> (URL of DSpace statistics page)

Instructions:

1. DSpace comes with a very basic web statistics package which you can use to gather information on page views (*not the same as file downloads*) and general activities performed within DSpace (e.g. searches performed, logins, errors and warnings). This statistics package will analyze and parse all of your old DSpace logs (in `[dspace]/log`) to generate HTML statistical reports. It consists of a set of Perl scripts all beginning with “stat-” in your `[dspace]/bin/` directory. (Note: In order to utilize these scripts you must have Perl installed!)
2. There are two basic options in `dspace.cfg` which allow you to specify whether or not your final statistics reports should be publicly accessible (`report.public`), and where the final statistical reports should be stored (`report.dir`). It's very possible you may not want to change either of these. However, if you change either option, don't forget to restart Tomcat (See *Quick Restart* in **Rebuild DSpace**)

```
report.public = false
report.dir = ${dspace.dir}/reports/
```

Use DSpace Statistics

(continued)

3. The DSpace Statistics package comes with its own configuration file, `dstat.cfg`. This file has some settings which you *must* change (marked with an asterisk below), and a few that you may wish to review:
 - `dspace.log*` – the full path of your `[dspace]/log` directory
 - `general.summary` – actions listed in the DSpace log file which you want to list in the “Overview” section. You don't need to change these unless needed.
 - `exclude.word` – stopwords to filter out of search terms in statistics
 - `exclude.type` – Lucene search index terms to filter out of statistics (corresponds to the search indices, see [Modify search fields](#)).
 - `exclude.character` – Lucene special characters to filter out of statistics
 - `item.type` – Item types to find statistics for. Corresponds to the form values (see [Change a form value](#)) defined for your `dc.type` field, or any metadata field with an element named `type`.
 - `item.floor` and `search.floor` – specify the number of minimum accesses necessary before an item or search term is listed in statistics
 - `item.lookup` – specifies the maximum number of items to list Author/Title information for in statistics (all other viewed items are listed by URL)
 - `user.email` – specifies whether to display user email information in login statistics. For privacy, this defaults to `false` (i.e. do not display email)
 - `host.name` and `host.url*` – The name and URL of your DSpace install, which will be displayed at the top of the statistics page.
4. In addition, the DSpace Statistics package comes with a mapping file (`dstat.map`), which maps DSpace “actions” which appear in the log files (in `[dspace]/log`) into human readable text. So, if you wish to update the text which appears in the statistics reports, or change the language, you can edit this file to do so.
5. First, you must modify each of these scripts slightly based on your own DSpace installation. Look for the following section in each of the `stat-*` files:

```
# Details used
#####
```

Use Dspace Statistics

(continued)

6. In that section, you will likely need to modify any variable which specifies the full path of a directory or file location (these paths will all start with `/dspace/` by default). In particular keep an eye out for these variables:
 - `$dsrun` – the full path of the `[dspace]/bin/dsrun` script
 - `$in_directory` – the full path of the input directory (for `stat-report-*` scripts)
 - `$out_directory` – the full path of the statistical reports output directory. This must correspond to the directory listed in the `report.dir` option in your `dspace.cfg` (see above)!
 - To keep things simple, you may wish to specify that both the `$in_directory` and `$out_directory` be the same location (e.g. `[dspace]/reports/`).
 - You may also wish to modify the `$start_year` and `$start_month` variables in `stat-initial` and `stat-report-initial`. These should correspond to the year and month which you wish to start tracking statistics from.
7. Make sure you updated *every* one of the `stat-*` scripts! There's six of them total: `stat-general`, `stat-initial`, `stat-monthly`, `stat-report-general`, `stat-report-initial`, `stat-report-monthly`
8. You will first need to run the initialization script to gather data about all of the past months (back to the month and year specified in `$start_month` and `$start_year`). This script only needs to be run *once*, though you can rerun it if you needed to (in which case it would overwrite its past results). Its output is a set of historic data files which can be used by `stat-report-initial` to generate historic HTML reports.


```
| [dspace]/bin/stat-initial
```

(Note: For Windows, you may need to run `perl [dspace]/bin/stat-initial`)
9. Next, generate a series of monthly reports for all of this past data by running the corresponding `stat-report-*` script. Again, this script only needs to be run *once*, though you can rerun it as needed. Its output is historic monthly HTML reports.


```
| [dspace]/bin/stat-report-initial
```

(Note: For Windows, `perl [dspace]/bin/stat-report-initial`)

Use DSpace Statistics

(continued)

10. The other four scripts should be run at least on a monthly basis, though you may even want to schedule them to run on a nightly basis in order to keep your statistics up-to-date at all times. These scripts will generate a report based on the activities this current month (`stat-monthly` and `stat-report-monthly`) and a general report aggregating all activities in the history of the repository (`stat-general` and `stat-report-general`). As in the examples above, the `stat-report-*` scripts should always be scheduled to run *after* their corresponding `stat-*` script.
11. For Linux or Mac OSX, you can schedule the scripts to run by adding a `cron` entry similar to the following to the crontab for the user who installed DSpace:

```
30 0 * * * [dspace]/bin/stat-monthly
35 0 * * * [dspace]/bin/stat-general

30 1 * * * [dspace]/bin/stat-report-monthly
35 1 * * * [dspace]/bin/stat-report-general
```

(The above cron entry would schedule these scripts to run nightly. `stat-monthly` and `stat-general` would be scheduled for 12:30am and 12:35am respectively. While their corresponding report scripts would run at 1:30am and 1:35am respectively. *Note: You would need to change `[dspace]` to the full path of your DSpace installation directory.*)

12. For Windows, you should use Windows Scheduled Tasks to schedule those same commands at an appropriate time of day. Just remember to schedule `stat-monthly` and `stat-general` to run before their corresponding report scripts (similar to the Linux/Mac OSX example above). In addition, you may need to call Perl explicitly (e.g. `perl stat-monthly`)
13. After running all statistics scripts, your DSpace site's statistical reports will be available immediately at <http://web-address-to-my-dspace/statistics>

Use the OAI-PMH interface

Files:

- `[dspace-source]/build/dspace-oai.war`
- <http://web-address-to-my-dspace/dspace-oai> (OAI-PMH base URL)

Instructions:

1. Whenever you rebuild DSpace (see **Rebuild DSpace**), the OAI-PMH (<http://www.openarchives.org/pmh/>) interface is automatically rebuilt into the `dspace-oai.war`. In order to enable the OAI-PMH interface, you just need to copy this `dspace-oai.war` file into your `[Tomcat]/webapps/` directory. (It likely will already be there if you followed the installation directions completely).
2. If you've never used OAI-PMH before, there is a Beginner's Tutorial from the Open Archives Forum available at: <http://www.oaforum.org/tutorial/>
3. Here's a few basic OAI-PMH “verbs” (i.e. commands) you may wish to familiarize yourself with (*all example DSpace URLs should be on a single line*):
 - Identify – used to retrieve information about a repository
 | <http://web-address-to-my-dspace/dspace-oai/request?verb=Identify>
 - ListMetadataFormats – lists metadata formats supported for OAI-PMH
 | <http://web-address-to-my-dspace/dspace-oai/request?verb=ListMetadataFormats>
 - ListRecords – permits metadata harvesting of all or a selection of items
 | http://web-address-to-my-dspace/dspace-oai/request?verb=ListRecords&metadataPrefix=oai_dc
 - ListSets – lists the “sets” (i.e. DSpace Collections) available in a repository
 | <http://web-address-to-my-dspace/dspace-oai/request?verb=ListSets>
4. You may wish to register your Base URL (see *Files* list above) in the following locations, so that others may harvest your metadata and link back to your site:
 - DSpace OAI Listing: <http://wiki.dspace.org/index.php/OaiInstallations>
 - OAI-PMH Data Provider Registry: <http://www.openarchives.org/data/registerasprovider.html>
 - Directory of Open Access Repositories: <http://www.openoai.org/>
 - Registry of Open Access Repositories: <http://roar.eprints.org/>
 - OAIster: <http://www.oaister.org/>

Configure RSS feeds

Files:

- `[dspace]/config/dspace.cfg`
- `http://web-address-to-my-dspace/dspace-oai` (OAI-PMH base URL)

Instructions:

1. All RSS feed options are available in `dspace.cfg` under the section with the following heading:


```
##### Syndication Feed Settings #####
```
2. To enable RSS feeds for new/updated items in your DSpace installation, you need to set the following variable to `true`:


```
webui.feed.enable = false
```
3. You can modify the maximum number of items to display in your RSS feed:


```
webui.feed.items = 4
```
4. To help with performance, you can chose the number of feeds to cache (`size`) as well as the number of hours to cache a feed (`age`):


```
webui.feed.cache.size = 100
webui.feed.cache.age = 48
```
5. You can also chose the RSS formats you would like to offer a feed in. It's recommended to chose either RSS 1.0 or RSS 2.0 (or both):


```
webui.feed.formats = rss_1.0,rss_2.0
```
6. You need to decide if you want your RSS feed(s) to refer to items via their handle (e.g. `http://hdl.handle.net/123456789/1`) or via your local DSpace URL. By default, the RSS feed contains an item's handle (specified as `false` below). But, you can change it to reference your local DSpace URL by setting it to `true`

```
webui.feed.localresolve = false
```
7. You can also customize exactly which metadata fields your RSS feed will display for all items. The information about an item in RSS generally consists of a *Title*, a *Date* and a *Description*. You can map the *Title* and *Date* using the following two options:


```
webui.feed.item.title = dc.title
webui.feed.item.date = dc.date.issued
```

Each of these options only accepts a single metadata field. Although, you can specify an asterisk (*) as a qualifier, to include all qualified fields (so `dc.title.*` would include `dc.title`, `dc.title.alternative`, etc.)

Configure RSS feeds

(continued)

8. For the RSS *Description*, you can actually specify a list of fields (all of which will be displayed as part of the description of an item). The format is the same as mentioned for the RSS *Title* and *Date* above, but now you can provide multiple fields separated by commas. (REMEMBER: if you expand this option across multiple lines, you must include a backslash (\) at the end of each continued line, as in the example below)

```
webui.feed.item.description = dc.title, \  
dc.contributor.author, dc.contributor.editor, \  
dc.description.abstract, dc.description
```

9. Once you've specified all of your feed options, you must restart Tomcat (See *Quick Restart* in **Rebuild DSpace**) for the changes to take effect.
10. You should see one or more RSS feed buttons (similar to below) appear on your DSpace homepage, as well as each Community and Collection homepage. Clicking on one of these buttons will provide you with a feed of recent submissions at either the site level (on DSpace homepage) or at a Community or Collection level.



Configure checksum checker

Files:

- `[dspace]/bin/checker`
- `[dspace]/config/dspace.cfg`

Instructions:

1. DSpace now comes with a Checksum Checker script (`[dspace]/bin/checker`) which can be scheduled to verify the checksum of every item within DSpace. Since DSpace calculates and records the checksum of every file submitted to it, this script is able to determine whether or not a file has been changed (either manually or by some sort of corruption or virus). The idea being that the earlier you can identify a file has changed, the more likely you'd be able to recover it (assuming it was not a wanted change).
2. There are several configuration options for the Checksum Checker which appear in the following section of `dspace.cfg`:

```
##### Checksum Checker Settings #####
```

3. The options you should most pay attention to are those regarding the checksum retention history (shown below). These two options specify how long a single checksum verification action is kept within your DSpace database. More information on each follows:

```
# check history retention
checker.retention.default=10y
checker.retention.CHECKSUM_MATCH=8w
```

- The `checker.retention.CHECKSUM_MATCH` option specifies the timeframe after which a successful “match” will be removed from your DSpace database (defaults to 8 weeks). This means that after 8 weeks, all successful matches are automatically deleted from your database (in order to keep that database table from growing too large).
- The `checker.retention.default` option specifies the default timeframe after which *all* checksum checks are removed from the database (defaults to 10 years). This means that after 10 years, all successful or unsuccessful matches are removed from the database.
- You can specify any timeframe for either of these options. Valid timeframes include: seconds (`s`), minutes (`m`), hours (`h`), days (`d`), weeks (`w`), years (`y`)
- Please note, these retention settings are only used if you specify the `-p` option for the checker (see below). Otherwise, they are ignored!

Configure checksum checker

(continued)

4. If you changed any option in the `dspace.cfg`, you will need to restart Tomcat (See *Quick Restart* in [Rebuild DSpace](#)) for the changes to take affect.
5. The Checksum Checker script (`[dspace]/bin/checker`) also has several command line options to be aware of:
 - `-c 10` = Limited Count Mode (`-c`). This limits the check to only the next 10 bitstreams (the checker will always start where it left off). *(Recommended for larger repositories who may only want to check a portion of their repository each evening)*
 - `-d 2h` = Duration Mode (`-d`). This tells the checker to run for 2 hours. The same timeframes are available as mentioned above for the `dspace.cfg` options. *(Recommended for larger repositories who may only want to check a portion of their repository each evening)*
 - `-b 111 112` = Specific Bitstreams Mode (`-b`). This tells the checker to only look at the bitstreams with internal IDs of 111 and 112.
 - `-a 1234/12` = Specific Handle Mode (`-a`). This tells the checker to only check bitstreams within the Item/Collection/Community specified by that handle.
 - `-l` = Looping Mode (`-l` or `-L`). A lowercase L (`-l`) specifies to check every bitstream in the repository once. *(Recommended for smaller repositories who are able to loop through all their content in just a few hours maximum)* An uppercase L (`-L`) specifies to continuously loop through the repository *(not recommended for most repository systems)*
 - `-p` = Enable pruning (`-p`). Tells the checker to actually remove old results from the database based on the retention settings specified within `dspace.cfg`. *Without this option, the retention settings are ignored and the database table may grow rather large!*
6. You should schedule the Checksum Checker to run automatically, based on how frequently you backup your DSpace instance (and how long you keep those backups around for). The size of your repository is also a factor. For very large repositories, you may need to schedule it to run for an hour (e.g. `-d 1h` option) each evening to ensure it makes it through your entire repository within a week or so. Smaller repositories can likely get by with just running it weekly.
 - For Linux or Mac OSX, you can schedule it by adding a `cron` entry similar to the following to the crontab for the user who installed DSpace:

```
| 0 4 * * 0 [dspace]/bin/checker -d2h -p
```

© 2006-2007 by all Authors/Contributors listed

This work is licensed under Creative Commons Attribution-Noncommercial-Share Alike 3.0 License.

Configure checksum checker

(continued)

(The above cron entry would schedule the checker to run scripts to run every Sunday at 4am for 2 hours. It also specifies to “prune” the database based on the retention settings in `dspace.cfg`. *Note: You would need to change `[dspace]` to the full path of your DSpace installation directory.*)

- For Windows, you will be unable to use the `checker` shell script. Instead, you should use Windows Scheduled Tasks to schedule the following command to run at the appropriate time(s):

```
[dspace]/bin/dsrun.bat
org.dspace.app.checker.ChecksumChecker -d2h -p
```

(The above command should appear on a single line.)

7. Optionally, you may choose to receive **automated emails** listing the Checksum Checkers' results. There is no shell script for this functionality, but it's still a rather easy change. Just make sure to schedule it to run *after* the checker has completed its processing (otherwise the email may not contain all the results).

- There are a few options you can specify to this email script:
 - `-a` = send all results (everything specified below)
 - `-d` = only report on deleted bitstreams
 - `-m` = only report on missing bitstreams
 - `-c` = only report on bitstreams whose checksums changed
 - `-u` = only report on un-checked bitstreams
 - You can also combine options (e.g. `-m -c`) for combined reports
- For Linux or Mac OSX, you can add another `cron` entry similar to the following to the crontab for the user who installed DSpace:

```
0 8 * * 0 [dspace]/bin/dsrun
org.dspace.checker.DailyReportEmailer -a
```

(The above cron entry would schedule the email to be sent at 8am each Sunday, reporting all the results. The above command should appear on a single line.)

- For Windows, you can use the same general command. However, you obviously should use Windows Scheduled Tasks to schedule it:

```
[dspace]/bin/dsrun.bat
org.dspace.checker.DailyReportEmailer -a
```

(The above command should appear on a single line.)

Troubleshoot an error

Directories/files:

- DSpace log file: `[dspace]/log/dspace.log` (usually)
- Tomcat log directory: `[Tomcat]/logs` (usually)
- `[dspace-source]/jsp/local/`
- Servlet directory: `[dspace-source]/src/org/dspace/app/webui/servlet`

Websites:

- DSpace technical FAQ: <http://wiki.dspace.org/TechnicalFaq>
- DSpace Sourceforge page: <http://sourceforge.net/projects/dspace/>
- Search dspace-tech mailing list:
<http://sourceforge.net/mailarchive/forum.php?forum=dspace-tech>

Instructions:

1. If the expected information simply never appears, leaving a blank browser screen, the problem is almost certainly with a servlet. The error message should be in either `[dspace]/log/dspace.log` or `[Tomcat]/logs/`
2. If you get an Internal Server Error, a JSP or some source code could be the problem. If your email address is set up in `dspace.cfg` as the DSpace administrator, check your email for the error listing. If not, `ssh` to the DSpace machine, `cd` to the log directory and `tail -100 dspace.log`.
 - (Windows) Go to the DSpace log directory, and open `dspace.log` in your favorite text editor. Look near the bottom for the error message.
3. Usually, the culprit file is in the first five to ten lines of the error stack listing. The error stack starts with a line similar to:


```

      yyyy-mm-dd time WARN ...           (OR)
      yyyy-mm-dd time ERROR ...
      
```
4. Search the DSpace technical FAQ and the dspace-tech mailing list for information on the error. Key words in the error often are useful search terms.
5. If all else fails, email a description of the error along with the error stack listing to dspace-tech@lists.sourceforge.net.

Customizing Manakin, Version 1.1

Rebuild Manakin

Directories:

- `[dspace-source]`
- `[manakin-source]`
- `[Tomcat]/webapps/` (Mac OS X Server: `/library/jboss/3.2/deploy`)
 - Note: JBOSS comes pre-installed with Mac OS X server. However, for both sever and desktop editions Tomcat may be used as with other platforms.

Quick Restart *(Just restarts the web server after configuration changes)*

1. Stop Tomcat (WARNING: this will bring down the website)
 - (Linux / OS X / Solaris) `[Tomcat]/bin/shutdown.sh`
 - (Mac OS X Server) Use Server Admin to stop Tomcat ("Application Server")
 - (Windows) Use Tomcat Service Monitor (in Notification Area) to stop Tomcat
2. Start Tomcat
 - (Linux / OS X / Solaris) `[Tomcat]/bin/startup.sh`
 - (Mac OS X Server) Use Server Admin to start Tomcat ("Application Server")
 - (Windows) Use Tomcat Service Monitor (in Notification Area) to start Tomcat

Quick Build: *(Quick build after smaller changes)*

1. Logon to the server DSpace / Manakin is running on (e.g. ssh). Make sure to login as the user who initially installed DSpace / Manakin!
2. Open a command prompt (if you don't have one already), and `cd [manakin-source]`
3. `ant -Ddspace=[manakin-source] build_wars` (recompiles all Manakin code and builds a new .war file)
4. `cp build/*.war [Tomcat]/webapps/`
 - (Mac OS X Server) `cp build/*.war /library/jboss/3.2/deploy`
5. Test changes in DSpace / Manakin

Rebuild Manakin

(continued)

Full Refresh/Rebuild: *(Completely refresh all of Manakin)*

1. Logon to the server DSpace / Manakin is running on (e.g. ssh). Make sure to login as the user who initially installed DSpace / Manakin!
2. Open a command prompt (if you don't have one already), and
`cd [manakin-source]`
3. `ant clean` (removes old compiled code)
4. `ant -Ddspace=[dspace-source] build_wars` (recompiles all Manakin and builds a new fresh .war file)
5. Stop Tomcat (*WARNING: this will bring down the website*)
 - (Linux / OS X / Solaris) `[Tomcat]/bin/shutdown.sh`
 - (Mac OS X Server) Use Server Admin to stop Tomcat ("Application Server")
 - (Windows) Use Tomcat Service Monitor (in Notification Area) to stop Tomcat
6. `cp build/*.war [Tomcat]/webapps/`
 - (Mac OS X Server) `cp build/*.war /library/jboss/3.2/deploy`
7. (Optional, but recommended) Completely remove any `manakin` or `dspace` directories created in `[Tomcat]/webapps` (`/library/jboss/3.2/deploy` for Mac OS X Server) - BE CAREFUL...you don't want to remove the wrong directory!
 - To force Tomcat to recompile everything, you may also wish to remove any `manakin` or `dspace` directories created in `[Tomcat]/work/Catalina/localhost`
8. Start Tomcat
 - (Linux / OS X / Solaris) `[Tomcat]/bin/startup.sh`
 - (Mac OS X Server) Use Server Admin to start Tomcat ("Application Server")
 - (Windows) Use Tomcat Service Monitor (in Notification Area) to start Tomcat
9. Test your changes in DSpace / Manakin

Install a theme

Files:

- `[new-theme-source]` (This where the new theme's source code is)
- `[manakin-source]/themes/[theme-dir]` (This will be the new location for the theme's source)
- `[dspace]/config/xmlui.xconf`

Instructions:

1. If you are installing a brand new theme that is not distributed with Manakin then preform the following operations, otherwise skip to the next step.

```
| cp [new-theme-source] [manakin-source]/themes/[theme-dir]
```

2. Open Manakin's configuration file, `[dspace]/config/xmlui.xconf`, scroll to the bottom and locate the `<themes>` element.

3. Add a new tag `<theme>` element inside the `<themes>` element as below:

```
| <theme name="My new theme" path="[theme-dir]"/>
```

4. If you wish to install this theme onto a particular community or collection then modify the new `<theme>` tag as:

```
| <theme name="My new theme" handle="123/1" path="[theme-dir]"/>
```

5. If you wish to install this theme as the *default* theme then modify your new `<theme>` tag (*make sure this tag is the last theme defined*):

```
| <theme name="My new theme" regex=".*" path="[theme-dir]"/>
```

6. Restart Tomcat (See *Quick Restart* in [Rebuild Manakin](#))

Notes:

- Themes are evaluated in the order they appear in the `xmlui.xconf` file. Thus, if the default theme is defined before any other themes then no other themes will be used.

Create a new theme

Files:

- `[manakin-source] /themes/[theme-dir]/sitemap.xmap` (The theme's configuration)
- `[manakin-source] /themes/[theme-dir]/style.css` (CSS stylesheet)
- `[manakin-source] /themes/[theme-dir]/style-ie.css` (CSS stylesheet for IE)

Instructions:

1. First determine a directory name for your new theme inside the `[manakin-source]/themes/` directory; this name will be referenced as `[theme-dir]` in these instructions.
2. Instead of starting your new theme completely from scratch, make a new copy of the *standard theme template* in a new directory, `[theme-dir]`.
 - `cd [manakin-source]/themes`
 - `cp -R template [theme-dir]`
5. Next customize the theme's sitemap by specifying the theme's directory and name. Open, `[manakin-source]/themes/[theme-dir]/sitemap.xmap`, and edit the global variables shown below:

```
<map:component-configurations>
  <global-variables>
    <theme-path>[theme-dir]</theme-path>
    <theme-name>[the name of your theme]</theme-name>
  </global-variables>
</map:component-configurations>
```

6. Next customize the theme's CSS stylesheets, by default there are two style sheets used – a base version for all browsers and then a supplemental version just for Internet explorer.

`[manakin-source]/themes/[theme-dir]/style.css`

`[manakin-source]/themes/[theme-dir]/style-ie.css`

7. Perform the steps in [Install a theme](#)
8. Perform the steps in [Rebuild Manakin](#)

Customize header or footer

Files:

- `[manakin-source]/themes/[theme-dir]/template.xml` (The theme's stylesheet)
- `[manakin-source]/themes/dri2xhtml/structural.xml`

Instructions:

1. If you haven't already, you need to [Create a new theme](#)
2. To modify the header, locate the template named “buildHeader” inside the structural.xml. Copy this template into your theme's local template.xml and modify the XSLT as needed.

```
<xsl:template name="buildHeader">
    ...
</xsl:template>
```

3. To modify the footer, locate the template named “buildFooter” inside the structural.xml. Copy this template into your theme's local template.xml and modify the XSLT as needed.

```
<xsl:template name="buildFooter">
    ...
</xsl:template>
```

Modify item metadata display

Files:

- `[manakin-source]/themes/[theme-dir]/template.xml` (The theme's stylesheet)
- `[manakin-source]/themes/dri2xhtml/DS-METS-1.0-DIM.xml` (DIM metadata handler)

Instructions:

1. If you haven't already, you need to [Create a new theme](#)
2. Locate the template named "itemSummaryView_DS-METS-1.0-DIM" inside the base DIM (DSpace Intermediate Metadata) handler. This is the template that is used to display an item when viewing it directly (another template is called when the user clicks show full item record).

```
<xsl:template name="itemSummaryView_DS-METS-1.0-DIM">
    ....
</xsl:template>
```

3. Copy the template identified above into your theme's local template.xml stylesheet. This will allow the theme to override the default behavior.
4. To modify the copied template to add a new Dublin Core field, add the following table row inside the `<table class="ds-includeSet-table">` element in the desired position. Here is an example:

```
<tr class="ds-table-row">
  <td><span class="bold">[Your label]</span></td>
  <td>
    <xsl:copy-of select="
      $data/dim:field[@element='[dc element]' and
      @qualifier='[dc qualifier]']/child::node()"
    />
  </td>
</tr>
```

5. To modify the copied template to remove a Dublin Core field, remove the (or comment out) the table row, `<tr>`, containing the unwanted field.

Notes:

- If the dublin core field you are adding does not have a qualifier, such as dc.type, then replace `"@qualifier='[dc qualifier]'"` with `"not(@qualifier)"` so that the rule does not match fields with a qualifier.

Add logos to an item's display

Files:

- `[manakin-source]/themes/[theme-dir]/template.xsl` (The theme's stylesheet)
- `[manakin-source]/themes/dri2xhtml/DS-METS-1.0-DIM.xsl`
(DIM metadata handler)

Instructions:

1. If you haven't already, you need to **Create a new theme**
2. Locate the template named "itemSummaryList_DS-METS-1.0-DIM" inside the base DIM (DSpace Intermediate Metadata) handler. This is the template that is used to display an individual item with in a larger list, as on the browse by page and search results.

```
<xsl:template name="itemSummaryList_DS-METS-1.0-DIM">
    . . .
</xsl:template>
```

3. Copy the template identified above into your theme's local template.xsl stylesheet. This will allow the theme to override the default behavior.
4. Modify the template to add an `` element for the logo. The image used will be based upon a Dublin Core field, in the example below `dc.type` is used.

```
<img class="item-logo">
  <xsl:choose>
    <xsl:when test="$data/dim:field[@element='type']='image'">
      <xsl:attribute name="src">
        <xsl:value-of select="$theme-path"/>
        <xsl:text>/images/image.jpg</xsl:text>
      </xsl:attribute>
    </xsl:when>
    ... add other icon cases ...
    <xsl:otherwise>
      <xsl:attribute name="src">
        <xsl:value-of select="$theme-path"/>
        <xsl:text>/images/unknown.jpg</xsl:text>
      </xsl:attribute>
    </xsl:otherwise>
  </xsl:choose>
</img>
```

Add text to a page (i18n)

Files:

- *[manakin-source]*/config/i18n/messages.xml (English i18n catalogue)
- *[manakin-source]*/config/i18n/messages*[language & locale]*.xml
(non-English i18n catalogues)

Instructions:

1. Open the theme's XSL template and modify the relevant templates (i.e. the header or footer templates) adding an `<i18n:text>` element where you would like the new text to appear. Make sure the value of the key is unique, for example:

```
| <i18n:text> xmlui.yourtheme.mynewkey </i18n:text>
```

2. Open the messages catalogue, config/i18n/messages.xml, and for neatness find the key values for that begin with the common prefix of your new key. Add a new line corresponding to your new key:

```
| <message key="xmlui.yourtheme.mynewkey">
|   Text of my new key!
| </message>
```

3. Perform the steps in **Rebuild Manakin**.

Notes:

- i18n = internationalization
- You can, of course, simply add the text directly into the XSL, but you will find it easier to translate the text into other languages if you use the i18n catalogue.
- Remember, when adding or modifying text in the messages.xml catalogues be very careful to use XML escaping rules, such as:

&	becomes	&
"	becomes	"e;
'	becomes	'
<	becomes	<
>	becomes	>

Install an Aspect

Files:

- `[new-aspect-source]` (This where the new Aspect's source code is)
- `[manakin-source]/aspects/[aspect-dir]`
(This will be the new location for the Aspect's source)
- `[dspace]/config/xmlui.xconf`

Instructions:

1. If you are installing a brand new Aspect that is not distributed with Manakin then preform the following operations, otherwise skip to the next step.

```
| cp [new-aspect-source] [manakin-source]/aspects/[aspect-dir]
```

(the above copy command should be on a single line)
2. Open Manakin's configuration file, `[dspace]/config/xmlui.xconf`, scroll to the bottom and locate the `<aspects>` element.
3. Add a new `<aspect>` element inside the `<aspects>` element as below:

```
| <aspect name="My new Aspect" path="[aspect-dir]"/>
```
4. Restart Tomcat (See *Quick Restart* in **Rebuild Manakin**)

Troubleshoot an error

Directories/files:

- DSpace log file: `[dspace]/log/dspace.log` (usually)
- Tomcat log directory: `[Tomcat]/logs` (usually)
- Cocoon logs: `[tomcat]/webapps/[webapp]/WEB-INF/logs/` (need to be turned on)

Websites:

- Manakin website: <http://di.tamu.edu/projects/xmlui/>
- Search dspace-manakin mailing list:
<http://sourceforge.net/mailarchive/forum.php?forum=dspace-manakin>

Instructions:

1. If the expected information simply never appears, leaving a blank browser screen, then the problem is likely with Tomcat's configuration. The error message should be in `[Tomcat]/logs/`
2. If you receive an "Invalid Continuation ID" error it means that you've restarted your Tomcat server and try to resume your session with the server. Re-login to DSpace / Manakin
3. If you need more logging information you can turn on Cocoon's logging, edit the `logkit.xconf` file. Change the sitemap's log-level to DEBUG as shown below:

```
| <category name="sitemap" log-level="DEBUG">
```
4. If all else fails, email a description of the error along with the error stack listing to dspace-manakin@lists.sourceforge.net.